



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/035,604	10/26/2001	Antti Huima	35997-215548	3832
23973 7590 12/10/2007 DRINKER BIDDLE & REATH ATTN: INTELLECTUAL PROPERTY GROUP ONE LOGAN SQUARE 18TH AND CHERRY STREETS PHILADELPHIA, PA 19103-6996			EXAMINER NGUYEN, THUONG	
			ART UNIT 2155	PAPER NUMBER
			MAIL DATE 12/10/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/035,604

Applicant(s)

HUIMA, ANTTI

Examiner

Thuong (Tina) T. Nguyen

Art Unit

2155

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 05 November 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 31-60 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 31, 40 and 42-60 is/are rejected.
- 7) ☒ Claim(s) 41 is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- ☒ Notice of References Cited (PTO-892)
- ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____
- ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- ☐ Notice of Informal Patent Application
- ☐ Other: _____

DETAILED ACTION

1. This action is responsive to the amendment filed on 11/5/07. Claims 1-30 are canceled. Claims 31-60 are added. Claims 31-60 are pending and represent method for managing compiled filter code.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. Claims 42-46 are rejected under 35 U.S.C. 101 because the claimed invention of the claims 42-46 are directed to non-statutory subject matter. Claims 42-46 recited "a computer readable medium having computer readable instructions to instruct a computer to perform a method comprises: " which are adapted to perform some steps.

The computer program and the program are non-statutory as not being tangible embodied in computer readable medium in a manner so as to be executable, and also claimed that the computer program/programming execute in a computer or by a computer are descriptive material per se and are not statutory because they are not capable of causing functional change in the computer (See MPEP section 2106, Seventh Edition, Revision No. dated February 2000, at page 2100-10 and 2100-11).

Other dependent claims, which are not specifically cited above are also rejected because of the deficiencies of their respective parent claims.

Claim Objections

4. Claim 47 objected to under 37 CFR 1.75(c) as being in improper form because a multiple dependent claim 31-41. See MPEP § 608.01(n). Accordingly, the claim 47 not been further treated on the merits.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 41 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. It's unclear to the examiner *stated the unclear part*?

7. Regarding claims 41, the phrase "substantially" renders the claim indefinite because it is unclear whether the limitation(s) following the phrase are part of the claimed invention. See MPEP § 2173.05(d). Appropriate correction is required.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 31-33, 42-43, 46, 48-49, 56-57 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1.

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

10. As to claim 31, Stack teaches method for processing data packets and managing compiled packet filtering code, comprising:

processing the received packets according to the compiled code (figure 1-4; col 3, lines 55-65; Stack discloses that the method that processing the predetermined application program); and

But Stack failed to teach the claim limitation wherein compiling packet filtering code and writing the compiled code to a memory; receiving data packets for processing; managing the compiled code in a plurality of pieces.

However, Nikander teaches method and arrangement for implementing IPSEC policy management using filter code (see abstract). Nikander teaches the limitation wherein compiling packet filtering code and writing the compiled code to a memory (col 3, lines 55-65); receiving data packets for processing (figure 5); managing the compiled code in a plurality of pieces (col 7, lines 60-68).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Stack in view of Nikander so that the system would be able to compile filter code of large volume of data packets. One would be motivated to do so to

the system would be able to make a decision based on the communication between the filter code and policy managers.

11. As to claim 32, Stack and Nikander teach method as recited in claim 31, further comprising:

incrementally compiling at least one rule into at least one piece of compiled code by a rule compiling entity (col 7, lines 14-46; Stack discloses the method of compiling the program rule begins at a logical root rule and proceeds through the ordered set of rules);

transmitting said piece of compiled code from the rule compiling entity to a packet processing entity (figure 2 & 4; Stack discloses the method of transmission the program code or rule by the rule table, control table and process sequences);

pausing processing of packets by said packet processing entity (col 8, lines 5-23; Stack discloses the method of processing the syntax rule that specify higher level functions or required different operations in the generation of the code or by user defined; Therefore, the system would pause or run according to the user's preferences);

writing said piece of compiled code to the memory (col 8, lines 24-40; Stack discloses the method of storing the program code to the memory or storage); and

restarting processing packets by said packet processing entity (col 14, table I; Stack discloses the method of defining the next segment accordingly; Therefore, the program would run or stop depend on user's preferences or setting).

12. As to claim 33, Stack and Nikander teach method as recited in claim 31, further comprising:

incrementally compiling at least one rule into at least one piece of compiled code by a rule compiling entity (col 7, lines 14-46; Stack discloses the method of compiling the program rule begins at a logical root rule and proceeds through the ordered set of rules);

signaling from said rule compiling entity to a packet processing entity that a new piece of compiled code is compiled (col 14, table I; col 21, table II; Stack discloses the method of popup the call or display according to the user's preferences or setting);

signaling from said packet processing entity to said rule compiling entity that said packet processing entity is ready for said piece of compiled code to be stored (col 14, table I; col 21, table II; Stack discloses the method of popup the call or display according to the user's preferences or setting);

writing said piece of compiled code to said memory (col 8, lines 24-40; Stack discloses the method of storing the program code to the memory or storage); and

signaling from said rule compiling entity to said packet processing entity that said piece of compiled code is written to said memory (col 16, table II; Stack discloses the method of popup the call or display according to the user's preferences or setting).

13. As to claim 42, Stack teaches a computer readable medium having computer readable instructions to instruct a computer to perform a method comprising:

processing the received packets according to compiled packet filtering code in a memory (figure 1-4; col 3, lines 55-65; Stack discloses that the computer readable medium that processing the predetermined application program).

But Stack failed to teach the claim limitation wherein receiving data packets for processing; managing the compiled code in a plurality of pieces.

However, Nikander teaches the limitation wherein receiving data packets for processing (col 3, lines 55-65); managing the compiled code in a plurality of pieces (col 7, lines 60-68).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Stack in view of Nikander so that the system would be able to compile filter code of large volume of data packets. One would be motivated to do so to the system would be able to make a decision based on the communication between the filter code and policy managers.

14. As to claim 43, Stack and Nikander teach computer readable medium as recited in claim 42 wherein the method further comprises:

incrementally compiling at least one packet filtering rule into least one piece of compiled code (col 7, lines 14-46; Stack discloses the computer readable medium of compiling the program rule begins at a logical root rule and proceeds through the ordered set of rules); and

updating the memory with said at least one piece of compiled code (col 8, lines 24-40; Stack discloses the computer readable medium of updating the program code to the memory or storage).

15. As to claim 46, Stack and Nikander teach computer readable medium as recited in claim 42 wherein the medium further has a library of software routines for processing data packets (figure 1).

16. A computer readable medium having computer readable instructions to instruct a computer to perform a method according to any one of claims 31 to 41.

17. As to claim 48, Stack teaches a computer network node for processing data packets comprising:

means for processing the received packets according to compiled code in a memory (figure 1-4; col 3, lines 55-65; Stack discloses that the computer network node that processing the predetermined application program).

But Stack failed to teach the claim limitation wherein means for receiving data packets for processing (col 3, lines 55-65); means for managing the compiled code in a plurality of pieces.

However, Nikander teaches the limitation wherein means for receiving data packets for processing; means for managing the compiled code in a plurality of pieces (col 7, lines 60-68).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Stack in view of Nikander so that the system would be able to compile filter code of large volume of data packets. One would be motivated to do so to the system would be able to make a decision based on the communication between the filter code and policy managers.

18. As to claim 49, Stack and Nikander teach computer network node as recited in claim 48, further comprising:

means for incrementally compiling at least one packet processing rule and producing at least one piece of compiled code (col 7, lines 14-46; Stack discloses the computer network node of compiling the program rule begins at a logical root rule and proceeds through the ordered set of rules); and

means for updating the memory with said at least one piece of compiled code (col 8, lines 24-40; Stack discloses the computer network node of updating the program code to the memory or storage).

19. As to claim 56, Stack teaches a system for processing data packets and managing compiled packet filtering code, comprising:

means for processing the received packets according to the compiled code (figure 1-4; col 3, lines 55-65; Stack discloses that the system that processing the predetermined application program).

But Stack failed to teach the claim limitation wherein means for compiling packet filtering code and writing the compiled code to a memory; means for receiving data packets for processing; means for managing the compiled filter code in a plurality of pieces.

However, Nikander teaches the limitation wherein means for compiling packet filtering code and writing the compiled code to a memory (col 3, lines 55-65); means for receiving data packets for processing; means for managing the compiled filter code in a plurality of pieces (col 7, lines 60-68).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Stack in view of Nikander so that the system would be able to compile filter code of large volume of data packets. One would be motivated to do so to the system would be able to make a decision based on the communication between the filter code and policy managers.

20. As to claim 57, Stack and Nikander teach system as recited in claim 56, further comprising:

means for incrementally compiling at least one rule into at least one piece of compiled code (col 7, lines 14-46; Stack discloses the system of compiling the program rule begins at a logical root rule and proceeds through the ordered set of rules); and

means for updating the memory with said at least one piece of compiled code (col 8, lines 24-40; Stack discloses the system of updating the program code to the memory or storage).

21. Claims 34 & 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1, and further in view of Moberg, Patent No. 6,698,015 B1.

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

22. As to claim 34, Stack and Nikander teach method as recited in claim 31. But Stack and Nikander failed to teach the claim limitation wherein said pieces are code pages having a predetermined length.

However, Moberg teaches apparatus and method for improving performance of critical code execution (see abstract). Moberg teaches the limitation wherein said pieces are code pages having a predetermined length (figure 7).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Moberg so that the system would be able to compile codes from small code. One would be motivated to do so to provide a faster performance for critical code across software changes.

23. As to claim 40, Stack, Nikander and Moberg teach method as recited in claim 34 wherein each page of code is associated with a reference number for observing the order of the code pages (col 10, lines 16-26; col 11, lines 50 – col 12, lines 48; Stack discloses the method of running the program sequences and orderly).

24. Claims 35-39 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1, and Moberg, Patent No. 6,698,015 B1, and further in view of Herrel, Patent No. 5,301,287 .

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

25. As to claim 35, Stack, Nikander and Moberg teach method as recited in claim 34. But Stack, Nikander and Moberg failed to teach the claim limitation wherein shadow paging is used.

However, Herrel teaches user scheduled direct memory access using virtual addresses (see abstract). Herrel teaches the limitation wherein shadow paging is used (col 15, lines 38-55).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack, Nikander and Moberg in view of Herrel so that the program code runs without the necessity of executing instructions in the host processor. One would be motivated to do so to started running the program without further host processor intervention, and hence data transmission is handling without the need to execute further instructions in the host processor.

26. As to claim 36, Stack, Nikander, Moberg and Herrel teach method as recited in claim 35, further comprising:

processing received packets according to a first set of code pages (figure 1-4; col 3, lines 55-65; Stack discloses that the method that processing the predetermined application program);

creating a second set of code pages (figure 5; col 5, lines 43- col 6, lines 16; Stack discloses that the method that creating a set of program code);

initiating processing received packets according to said second set of code pages (col 10, lines 27-61; Stack discloses that the method that processing a set of program code); and

continuing processing packets received before said initiating, according to said first set of code pages (figure 3-4, 7; col 11, lines 50 – col 12, lines 50; Stack discloses that the system of processing program node accordingly).

27. As to claim 37, Stack, Nikander, Moberg and Herrel teach method as recited in claim 36, comprising within said step of creating a second set of code pages the steps of:

assigning one or more members of an existing code page set to be members of said second set of code pages (figure 5-7; col 8, lines 64 – col 9, lines 18; col 9, lines 40-65; Stack discloses that the method of validating the member of the program rules and segments); and

removing one or more code pages from said second set of code pages (figure 5-7; col 8, lines 64 – col 9, lines 18; col 9, lines 40-65; Stack discloses that the method of removing the member of the program rules and segments).

28. As to claim 38, Stack, Nikander, Moberg and Herrel teach method as recited in claim 36, comprising within said step of creating a second set of code pages the steps of:

creating a new code page (figure 5; col 5, lines 43- col 6, lines 16; Stack discloses that the method that creating a set of program code); and

assigning said new code page to be a member of said second set of code pages (figure 5-7; col 8, lines 64 – col 9, lines 18; col 9, lines 40-65; Stack discloses that the method of creating program code and checking for the set of rules).

29. As to claim 39, Stack, Nikander, Moberg and Herrel teach method as recited in claim 36, further comprising removing a code page from the memory when the code page is no longer a member of any set of code pages in use (figure 5-7; col 8, lines 64 – col 9, lines 18; col 9, lines 40-65; Stack discloses that the method of storing the program code and syntax and program rules).

30. Claims 44-45, 50-51, 58-59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1, and further in view of Herrel, Patent No. 5,301,287.

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

31. As to claim 44, Stack and Nikander teach computer readable medium as recited in claim 42. But Stack and Nikander failed to teach the claim limitation wherein implementing shadow paging of code pages.

However, Herrel teaches user scheduled direct memory access using virtual addresses (see abstract). Herrel teaches the limitation wherein implementing shadow paging of code pages (col 15, lines 38-55).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Herrel so that the program code runs without the necessity of executing instructions in the host processor. One would be motivated to do so to started running the program without further host processor intervention, and hence data transmission is handling without the need to execute further instructions in the host processor.

32. As to claim 45, Stack, Nikander and Herrel teach computer readable medium as recited in claim 44 wherein the method further comprises:

processing received packets according to a first set of code pages (figure 1-4; col 3, lines 55-65; Stack discloses that the computer readable medium that processing the predetermined application program);

creating a second set of code pages (figure 5; col 5, lines 43- col 6, lines 16; Stack discloses that the computer readable medium that creating a set of program code);

initiating processing received packets according to said second set of code pages (col 10, lines 27-61; Stack discloses that the computer readable medium that processing a set of program code); and

continuing processing packets received before said initiating, according to said first set of code pages (figure 3-4, 7; col 11, lines 50 – col 12, lines 50; Stack discloses that the computer readable medium of processing program node accordingly).

33. As to claim 50, Stack and Nikander teach computer network node as recited in claim 48. But Stack and Nikander failed to teach the claim limitation wherein implementing shadow paging of code pages.

However, Herrel teaches the limitation wherein implementing shadow paging of code pages (col 15, lines 38-55).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Herrel so that the program code runs without the necessity of executing instructions in the host processor. One would be motivated to do so to started running the program without further host processor intervention, and hence data transmission is handling without the need to execute further instructions in the host processor.

34. As to claim 51, Stack, Nikander and Herrel teach computer network node as recited in claim 50, further comprising:

means for processing received packets according to a first set of code pages (figure 1-4; col 3, lines 55-65; Stack discloses that the computer network node that processing the predetermined application program);

means for providing a second set of code pages (figure 5; col 5, lines 43- col 6, lines 16; Stack discloses that the computer network node that creating a set of program code);

means for initiating processing received packets according to said second set of code pages (col 10, lines 27-61; Stack discloses that the computer network node that processing a set of program code); and

means for continuing processing packets received before said initiating, according to said first set of code pages (figure 3-4, 7; col 11, lines 50 – col 12, lines 50; Stack discloses that the computer network node of processing program node accordingly).

35. As to claim 58, Stack and Nikander teach system as recited in claim 56. But Stack and Nikander failed to teach the claim limitation wherein implementing shadow paging of pages of compiled code.

However, Herrel teaches the limitation wherein implementing shadow paging of code pages (col 15, lines 38-55).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Herrel so that the program code runs without the necessity of executing instructions in the host processor. One would be motivated to do so to started running the program without further host processor intervention, and hence data transmission is handling without the need to execute further instructions in the host processor.

36. As to claim 59, Stack, Nikander and Herrel teach system as recited in claim 58, further comprising:

means for processing packets according to a first set of code pages (figure 1-4; col 3, lines 55-65; Stack discloses that the system that processing the predetermined application program),

means for creating a second set of code pages (figure 5; col 5, lines 43- col 6, lines 16; Stack discloses that the system that creating a set of program code);

means for initiating processing received packets according to said second set of code pages (col 10, lines 27-61; Stack discloses that the system that processing a set of program code); and

means for continuing processing packets received before said initiating, according to said first set of code pages (figure 3-4, 7; col 11, lines 50 – col 12, lines 50; Stack discloses that the system of processing program node accordingly).

37. Claims 52-55 are rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1, and further in view of Wesinger, Patent No. 6,052,788.

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

38. As to claim 52, Stack and Nikander teach computer network node as recited in claim 48. But Stack and Nikander failed to teach the claim limitation wherein the node is a virtual private network node.

However, Wesinger teaches firewall providing enhanced network security and user transparency (see abstract). Wesinger teaches the limitation wherein the node is a virtual private network node (figure 1).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Wesinger so that * the system could process remotely. One would be motivated to do so to broaden the flexibility of the system.

39. As to claim 53, Stack and Nikander teach computer network node as recited in claim 48. But Stack and Nikander failed to teach the claim limitation wherein the node is a router node.

However, Wesinger teaches the limitation wherein the node is a router node (figure 4).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Wesinger so that the packet could transfer from one machine to another in the WAN network. One would be motivated to do so to connect over a wide geographical area via one or more dedicated connections.

40. As to claim 54, Stack and Nikander teach computer network node as recited in claim 48. But Stack and Nikander failed to teach the claim limitation wherein the node is a firewall node.

However, Wesinger teaches the limitation wherein the node is a firewall node (figure 1).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Wesinger so that the system would maintain the security for the system. One would be motivated to do so to ensure the security for the processing process.

41. As to claim 55, Stack and Nikander teach computer network node as recited in claim 48. But Stack and Nikander failed to teach the claim limitation wherein the node is a workstation.

However, Wesinger teaches the limitation wherein the node is a workstation (figure 5).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Wesinger so that the client could oversee what is going on between the systems. One would be motivated to do so to broaden the flexibility of the system.

42. Claim 60 is rejected under 35 U.S.C. 103(a) as being unpatentable over Stack, Patent No. 6,257,774 B1 in view of Nikander, Patent No. 6,253,321 B1, and further in view of Kloth, Patent No. 6,598,034 B1.

Stack teaches the invention substantially as claimed including application program and documentation generator system and method (see abstract).

43. As to claim 60, Stack and Nikander teach system as recited in claim 56. But Stack and Nikander failed to teach the claim limitation wherein the memory has a first access port and a second access port, the system further comprising means for processing data packets, said processing means being arranged to access said memory via said first access port, and said means for managing the compiled filter code being arranged to access said memory via said second access port.

However, Kloth teaches rule based IP data processing (see abstract). Kloth teaches the limitation wherein processing data packets (figure 5), said processing means being arranged to access said memory via said first access port (figure 11), and

said means for managing the compiled filter code being arranged to access said memory via said second access port (figure 7).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the combination of Stack and Nikander in view of Kloth so that the system could processing the IP packets by dividing the packet into plurality pieces and process the packet one at a time by the communication protocol stack and the communications rule editor. One would be motivated to do so to broaden the flexibility of the system and applied the specific rule to specific packet to fasten the processing process.

Allowable Subject Matter

44. Claims 34, 40-41 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

45. The following is an examiner's statement of reasons for objected the claims :
In interpreting the claims, in light of the specification and the applicant's arguments filed on 11/5/07, the Examiner finds the claimed invention to be patentably distinct form the prior art of record.

46. Stack et al. (US 6,257,774 B1), teach application program and documentation generator system and method wherein processing the received packets according to the compiled code (abstract; figure 1-4; col 3, lines 55-65).

47. Nikander et al. (US 6,253,321 B1), teach method and arrangement for implementing IPSSEC policy management using filter code, wherein compiling packet filter code and writing code to a memory (abstract; col 3, lines 55-65).

48. The following is an examiner's statement of reasons for objected the claims to be allowed:

49. The examiner has found that the prior art of record does not appear to teach or suggest or render obvious the claimed limitations in combination with the specific added limitations as recited in dependent claims. The prior art of record fails to teach or suggest individually or in combination of the order of any two code pages is determined by comparing values of $v(x)$ calculated from the reference numbers associated with the code pages, $v(x)$ being calculated substantially by the formula

$$v(x) = r(x) - r(b) \text{ mod } M$$

where $r(x)$ is the reference number associated with a code page x being compared, $r(b)$ is the reference number of a base code page, and M is the size of a set of allowed reference numbers (0, 1, 2, ..., $M-1$).

Claims 34, 40-41 are object to be allowed because of the combination of other limitations and the limitation listed above.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tina Nguyen whose telephone number is 571-272-3864, and the fax number is 571-273-3864. The examiner can normally be reached on 8:00 AM-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Saleh Najjar can be reached on 571-272-4006. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Application/Control Number:
10/035,604
Art Unit: 2155

Page 23

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Thuong (Tina) Nguyen
Patent Examiner/Art Unit 2155

Philip Tran
PHILIP TRAN
PRIMARY EXAMINER